CS29003 ALGORITHMS LABORATORY
(Tutorial 4 : Trees – Solution)
Date: Sep 26 2020

1 Binary Tree to BST with Minimum Swaps

Solution -
Inorder traversal of Binary Search Tree is in increasing order of their value. So, find the inorder traversal of the Binary Tree and store it in the array and try to sort the array. The minimum number of swap required to get the array sorted will be the answer.

2 K^{th} Largest Element in BST

Solution -
1. Do reverse inorder traversal (RIT) of BST i.e., visit the right node, then centre node, and then left node.
2. RIT traverses all nodes in decreasing order. While doing the traversal, keep track of the count of nodes visited so far.
3. When the count becomes equal to k, stop the traversal and print the key.
Traversing to the rightmost node takes $O(h)$ time, where h is the height of the. Finding the k^{th} element takes $O(k)$. Overall time complexity is $O(h + k)$.

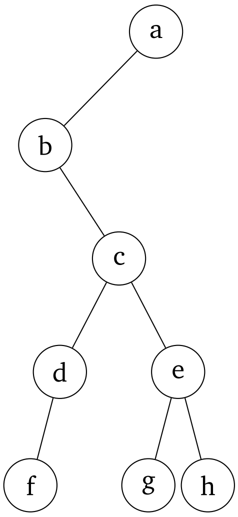
3 Maximum Weight

Solution-

```
maxweight (T, wt)
if (T == NULL) return wt
wt += T → val /* Add the value stored at the current node */
lwt = maxweight(T →L,wt) /* Recurse on the left subtree */
rwt = maxweight(T →R,wt) /* Recurse on the right subtree */
/* Take the larger of the weights returned by the two recursive calls */
wt = max(lwt,rwt)
return wt
```

4 Reconstruction of Tree

Solution-
The last element in the postorder listing is the root, so T has root a. Since nothing follows a in the inorder listing, the right subtree of a is empty. We need to construct the left subtree of T from the inorder listing bfdcgeh and postorder listing fdghecb. The root is b. Since nothing precedes b in the inorder listing, the left subtree of b is empty. So we recursively construct the right subtree of b from the inorder and postorder listings fdcgeh.The final tree is shown below.



5 Optimal sequence for AVL tree insertion

Solution-
1. Sort the given array of integers.
2. Create the AVL (BST) tree from the sorted array.
a) Get the middle element and make it root.
b) Recursively construct the left subtree and the right subtree.
3. Find the level order traversal of the tree which is the required sequence. Adding numbers in this sequence will always maintain the height balance property of all the nodes in the tree.